关于多目标道路寻优问题的研究

摘要

本文研究了交通网络的多目标道路寻优问题。首先将资料中影响路线选择的因素进行分析，并最终确定了三个优化目标：出行时间、出行费用、换乘次数。结合实际情况，在选择最优路线时，以出行时间为主要评价标准，将出行费用和换乘次数作为次要评价标准。将公交网络问题整合成图论问题，再对三个问题建立不同的模型，运用 matlab7.0 编程求出合理的解决方案。

对问题一，采用传统的 Dijkstra 算法无法处理好题中给出的海量数据，此外，考虑到公交换乘问题，我们提出了两个模型：第一个是基于 Dijkstra 算法的网络疏松化模型。在公交网络图中，边权不仅由相邻两结点的通行耗时产生，还会在结点内部产成（即换乘耗时）。为此，我们提出疏松处理方法，将结点内部耗时统一为边权，使模型与经典的图论最短路问题一致。然后，考虑到实际出行中换乘次数不会过多，提出第二个模型--换乘次数受限的通用模型：将换乘次数上限定为 3，按换乘次数 k（k=0,1,2,3）将乘车路线分类，分别求出各类中的最优解，通过比较得出最优乘车路线。

解决第二问时，沿用第一问的分类思想，提出子问题定理，建立分类拓展的限制换乘模型。为此，我们将路线分为：不经过地铁、仅在地铁站进行公交换乘和乘坐地铁这三类路线。第一类路线直接用问题一中的模型求解。对于第二、三类路线，分析题中所给条件并进行适当假设可得出：最短路径上有且仅有一次公汽换乘发生在地铁站，乘客只能进出地铁站一次。这样就将第二、三类路线分解成为由起始点到地铁站，再由地铁站到目的地的子问题，这些子问题可直接用第一问中的模型求解。分别对子问题求最优路径，可得到全程最优路径。

对问题三，由于考虑步行换站，原先的交通图拓展为完全交通图，将问题一与问题二的模型拓展应用，使该问题得到合理的分析与解决。

在求解过程中，我们采用记录并调用中间结果的方法，大大提高了运算效率。此外，我们根据得出的结果进行了灵敏度分析，总结出当公车行驶速度和换乘等待时间变化时，最短出行时间和出行选择的变化规律，并给乘客提出一些意见。最后，我们采用遗传算法对模型进行改进。

关键词： 多目标综合决策；疏松处理；子问题定理；换乘次数受限；遗传算法

1. 问题重述

公共交通系统是城市交通系统的重要组成部分。由于对交通资源的大量需求，实行公交优先成为缓解日趋严重的道路交通紧张状况的必然选择。

随着 08 年北京奥运会的到来，大量流动人口将涌入北京，使本就接近饱和的交通系统负担更加沉重。基于北京公交系统的繁重现状，人们在出行时面临着多条线路的择优问题。在路径选择时，乘客会综合出行时间、出行费用和换乘次数等因素作出决定。针对市场需求，某公司准备研制开发一个解决公交线路选择问题的自主查询计算机系统。系统的设计核心是线路选择的模型与算法能满足查询者的各种不同实际需求。

本题给出了北京市公交系统信息（包括公汽和地铁）和人们各种出行方式的单位耗时等资料。根据这些资料，利用数学建模的方法，依据人们的出行需求，给出任意两公汽站点之间线路择优问题的解决方案。

2. 问题分析

本题是一个交通网络的路径寻优问题。按照传统的想法，乘客总是选择从起始点到终迄点的最短路径。而研究表明，在大部分的城市公交网络模型中，最优路径并不意味着最短，而是指在到达出行目的的众多线路中，能最好的满足出行者愿望的线路，即出行效用最大的线路。对城市公交线路进行最优路径分析，需要将题中给出的信息整合成网络图论中的网络图，然后通过图论中的网络分析理论来实现公交网络中的最优路径分析。

在公交网络中，所有站点是通过公交线络连接在一起的，可以将整个城市的所有站点看作是连通图上的点。从任一站点出发，经过有限次转车可以到达目的地。考虑到现实生活中，换乘次数、出行耗时、出行费用等因素都会影响人们的出行乘车方案。而人们的出行需求最终可归结为两种形式：一种是总出行时间最少，另一种是乘客的出行总成本最少。

综合以上因素，本文建立换乘次数受限的通用模型解决最优路径问题。

3. 模型假设和符号系统

3.1 模型假设

⑴交通系统看作有向拓扑图，公汽站点和地铁站点统称站点，在网络模型中都看作结点；将公汽线路和地铁线路统称线路，在网络模型中都看作边。

⑵乘客在乘车时，只考虑换乘次数、出行耗时、出行费用三种因素，而不考虑景观道路、公交系统承载能力、道路状况、满意程度、高峰择流等因素的影响。

⑶题中所给数据包含该市的所有公交信息，出行方式只有步行、公汽和地铁三种，不存在公交车临时停车、增线以及乘坐出租车、私车等情况。

⑷公交系统时刻正常运行，题中给出的行驶时间、换乘耗时和公交票价均不变，不存在交通堵塞、自然灾害、物价上涨等影响公交系统的情况。

⑸公汽票价只有单一票价和分段计价两种形式。无论乘客在地铁线路间是否换乘，乘车费用不变。同一地铁站对应的任意两个公汽站之间可以通过地铁站换乘，无需支付地铁费。

3.2 符号系统

L_i 公汽线路，i 表示公汽线路编号；

S_i 公汽站点，i 表示公汽站点编号；

l_{S_i} 经过站点 S_i 的公汽线路；

D_i 地铁站点，i 表示地铁站点编号；

$\mathrm{t}_{\mathrm{ij}}^{\mathrm{n}}$ 从第 i 站到第 j 站换乘 n 次车所需的乘车时间，$n = 0,1,2,......$；

t_{ij} 从第 i 站到第 j 站所需的乘车时间；

$M_{\mathrm{ij}}^{\mathrm{n}}$ 从第 i 站到第 j 站换乘 n 次车所需的乘车费用，$n = 0,1,2,......$；

M_{ij} 从第 i 站到第 j 站所需的乘车费用；

H 步行时间矩阵；

H_{ij} 步行时间矩阵 H 中的元素，表示从第 i 站到第 j 站步行时间；

$G(V,E)$ 由站点和线路组成的网络图，$V=(S \cup D)$ 表示站点集合，E 表示线路集合；

$\mathrm{E}_{(\mathrm{i,j})}$ 从第 i 站到第 j 站的线路。

4. 模型准备

对城市交通路线进行最优路径分析，需要将现实中的城市公交网络整合成为网络图论中的网络图，然后通过图论中的网络分析理论来实现道路的最优路径分析。分析如下：

1）公交网络拓扑结构：

将站点看作结点，线路看作边。建立一个储存公交网络拓扑结构的集合，通过它可以获得各个站点周围的公交信息和道路信息。以各个车站站点作为结点建结点集$\{\mathrm{S}\}$，相邻两站之间的通路构造边集$\{\mathrm{E}\}$。构造交通拓扑图$G(S,E)$，E_{ij}为有向边，表示车站S_i和S_j（i,j表示车站编号）之间有车直达。

2）公交网络的有向线性质：

起点和终点决定公交线路的行驶路径和方向，不同的公交线路有不同的行驶路线和

方向。即使同一路公交车，其上行线和下行线行驶的路径和站点也不可能完全重叠。在网络拓扑结构中，我们将线的边权设为两站之间的行车时间，用W_{ij}表示，由题可知W_{ij}＝3min。

3）评价最优路线的因素：

在实际情况中，人们选择路径不仅要考虑出行时间t、出行花费M，而且要考虑到达目的地的方便程度，比如换车次数N等。考虑实际中，人们首要目标是尽快到达目的地，同时花费尽量少，而换乘次数在能接受的范围内即可。在本文中，始终将时间t作为首要的优化目标，出行花费M作为次要目标或评价标准，换乘次数N作为限制条件建立模型。

5. 问题一

5.1 问题分析：

综合出行时间、换乘次数以及出行费用这三个因素，我们分别将出行时间和出行费用作为目标函数建立模型求解，结合两种结果评价出最优乘车路线。由于题中明确给出了各路段行驶时间、换乘时间和乘车收费标准，而且换车耗时较多并可能需要额外花费，所以在用时最少的情况下，相应的换乘次数和费用也不会很高。根据实际经验，我们首先以用时最少为目标建立模型。

5.2 模型一：基于Dijkstra算法的网络疏松化模型

本题与一般的最短路问题的区别在于乘车时间花费不仅与边权有关，而且与换乘耗时有关。于是我们采用疏松处理的方法，将换乘耗时与边权统一起来，使交通网络图整合成为连通图，再用经典的Dijkstra算法求解。

✧ 疏松处理方法：

⑴将G中的每一个站点用站点和公交线路的点集代替。

⑵新增点代表在同一站点有不同站牌，在不同站牌之间增加边，则换乘时间可看作边权。

⑶新的结点连接成的子图为一个完全图，子图的每一个边权为5min。

⑷若一个其它结点与原结点连通，则它与每一个新结点连通。

疏松处理之后，新结点构成集合$\tilde{V}$，其中一个结点表示为$V(S_i, L_j) \in \tilde{V}$，则上述疏松处理方法可表示为：

(1) $\tilde{E}$表示含有新结点的公交网络图边集，$\forall l_j, l_k \in l_{S_i}$；

(2) $E_{((S_i,L_j),(S_i,L_k))} \in \tilde{E}$，$W_{ij} = \Delta t_{ij}$；

(3) $E_{(S_i,L_m),(S_j,L_n)} \in \tilde{E}, \text{if} \quad E_{(S_i,S_j)} \in E$ 。

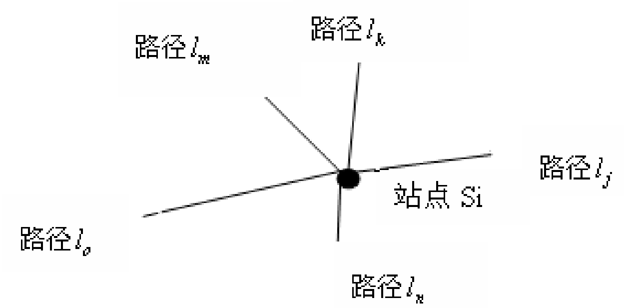


图1 疏松处理前的站点

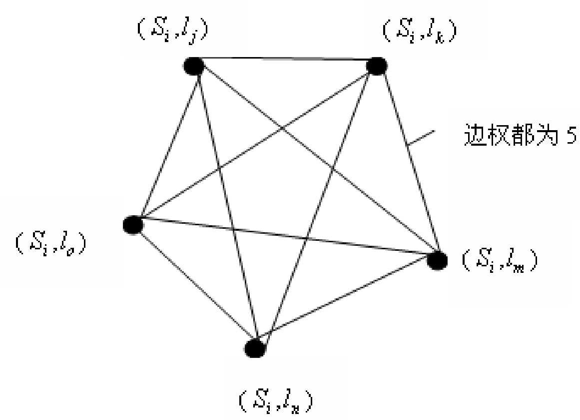


图2 疏松处理后的结点

由此可构造出新的拓扑图$\tilde{G}(\tilde{V},\tilde{E})$。那么经过处理之后，出行的时间花费就仅与边权有关，该问题就转化为一个非常典型的最短路问题，利用Dijkstra算法可求得结果。

但是该模型只分析了时间因素，而且由于题中的数据量庞大（车站点有3957个），经典的Dijkstra算法求解复杂度是O(N^3)，效率低下，难以求出全局最优解。因此根据实际情况，对题中一些条件进行限制，降低问题复杂度，建立新的模型求解。

5.3 模型二：基于换车次数受限的通用模型

5.3.1 模型补充：

在实际出行中，人们出于习惯和方便考虑，不会换乘太多的车次。那么我们不妨给定换乘次数的上限，从实际情况出发考虑，上限取3几乎可涵盖所有乘车情况。这样换乘次数k只有4种取值，即0，1，2，3，我们把N取不同值分别考虑。称所有经历k次换乘路线中的最短路为k阶最短路。这样低阶最短路的问题就成为高阶最短路问题的子问题。由此，可以得到一种较为简化的递归算法。

补充定义：

- $\{l_{si}\}$为站点S_i所属的所有线路构成的集合
- $m(j,k)$为在这条线路上站点j与k之间经过的站数
- 从起始站到终止站的时间花费的为t_{sd}，从S_S经历N次换车到S_d用的最少时

间为 t_{sd}^{N}

- {R_{sd}^{k}}是从 S_S 是经历N次换车到达 S_d 的路径序列， r_{sd}^{t} 是最短路径序列
- 从起始站到终止站的时间花费的为 $\mathrm{M_{sd}}$，从 S_S 经历N次换车到 S_d 用的最少时间为 $\mathrm{M_{sd}^{N}}$

✧ 定理一（子问题定理）：如果 S_k 是 S_S 到 S_d 最短路径上一点，那么这个最短路 R_{sd} 中 S_S 到 S_k 部分 R_{sk} 是 S_S 到 S_k 的最短路， S_k 到 S_d 部分 R_{kd} 是 S_k 到 S_d 的最短路。

证明（反证法）：如果存在 R'_{sk} 比 R_{sk} 短，则 R'_{sk} 和 R_{kd} 组成的路径比 R_{sd} 短，这与 R_{sd} 是 S_S， S_d 之间的最短路的假设矛盾。证毕。

在定理一的帮助下，可将k阶最短路问题分解成k-1阶最短路问题和0阶最短路问题：假定从 S_S 到 S_d 第一个换乘点为 S_p，则根据定理一，经过 S_p 的最短路由 S_S 到 S_p 的0阶最短路和 S_p 到 S_d 的k-1阶最短路组成。

假如k-1阶最短路径问题是可以解决的，那么就能通过枚举所有可能的 S_p 解出k阶最短路问题。因为0阶最短路问题可以解决，根据以上的分析得到了解决任意阶最短路问题的方法。

5.3.2 模型建立与求解：

换车次数受限的通用模型根据目标函数的不同可以用不同的形式给出：

◆ 以时间最省为目标：

- 目标函数：

$$\mathrm{Min} \quad \mathrm{t}_{sd} = \{\mathrm{t_{sd}^{N}}\} \quad , \quad \mathrm{N}=0,1,2,3$$

- 函数说明：

当N=0，若 $l_{ss} \cap l_{sd} \neq \varnothing$，则 $\mathrm{t_{sd}^{0}}=m(s,d)*3$；否则 $\mathrm{t_{sd}^{0}}=\infty$。

当N=k(k=1,2,3)，$t_{sd}^{k}=\min\{t_{sn}^{0}+5+t_{nd}^{k-1}\}$，n为第一次换乘的车站号，可 以从 S_S 向后沿各个线路顺次任意取站点。可用迭代的方法求出所有的 t_{sd}^{k} 。

- 算法步骤:

Step1:N=0(不换乘):以 S_S 为起始点广度优先搜索所有在{l_{S_i}}上的站点。如果找到 S_d,则 $\mathrm{t_{sd}^{0}}=\mathrm{m}(S_S,S_d)*3$，{$R_{sd}^{k}$}可直接找到；否则， $\mathrm{t_{sd}^{0}}=\infty$，{$R_{sd}^{k}$}为空。

Step2:N=1(换乘一次):以 S_S 为起始点广度优先搜索所有在{ l_{S_i} }上的站点 S_k， 返回Step1:

求出 t^0_{kd} , R^0_{kd} 。若 $t^0_{kd}=\infty$，则 $t^1_{sd}=\infty$；否则 $t^1_{sd}=t^0_{kd}+5+t^0_{sk}$ 。若 r^0_{kd} =null，则 r^1_{sd} =null；

否则{ r^1_{sd} }={ r^0_{kd} , r^0_{sk} }。

Step3:N=2(换乘两次):以 S_S 为起始点广度优先搜索所有在{ l_{S_i} }上的站点 S_k， 返回Step2

求出 t^0_{kd} , R^0_{kd} 。若 t^1_{kd} =无穷，则 $t^2_{sd}=\infty$；否则 $t^2_{sd}=t^0_{sk}+5+t^1_{kd}$ 。若 r^0_{kd} =null，则 r^2_{sd} =null；

否则{ r^2_{sd} }={ r^1_{kd}， r^0_{sk} }。

Step4: N=3(换乘三次):同理。

Step5: Min $t_{sd}=\{t^0_{sd},t^1_{sd},t^2_{sd},t^3_{sd}\}$， 比如 $t_{sd}=t^k_{sd}$, 则 $R_{sd}=r^k_{sd}$ 。算法终止。

- 模型求解及结果分析：

用matlab7.0编程进行搜索(程序见附录)，结果如下：

表 1：以时间为目标函数的求解结果

	最短时间(单位：min)					
换乘次数	出行 1	出行 2	出行 3	出行 4	出行 5	出行 6
0	∞	∞	∞	∞	∞	∞
1	101	∞	128	83	∞	65
2	64	106	103	67	106	46
3	69	99	105	63	102	51
任意换乘次数	64	99	103	63	102	46

说明：∞表示在该换乘次数下不存在一条线路可以到达终点站。

由答案可以看出换乘次数从 0 次到 2 次，乘车的最短时间有较大改善，而从 2 次到 3 次，6 组中只有 3 组的乘车时间会减少，并且减少的时间量很小，在换乘 2 次下得到的解已经十分接近最优解。又综合考虑增加换车次数会导致乘车费用增加，并且频繁换车有悖人的一般心理习惯，所以不再考虑换乘 3 次以上的乘车线路，处理较为合理。

按题中要求给出满足不同要求时的乘车路线:

表 2：以时间最少和换乘次数最少为目标的乘车路线

	用时最少的乘车路线	换乘次数最少的乘车路线
出行 1	S3359-L015-S2903-L485-S1784-L167-S1828	S3359-L436-S1784-L167-S1828
出行 2	S1557-L084-S1919-L189-S3186-L091-S0902-L254-S0481	S1557-L084-S1919-L189-S3186-L460-S0481
出行 3	S0971-L013-S2517-L290-S2159-L469-S0485	S0971-L013-S2184-L417-S0485
出行 4	S0008-L043-S1383-L002-S1327-L328-S0525-L103-S0073	S0008-L159-S0400-L474-S0073
出行 5	S0148-L308-S3604-L081-S2361-L156-S2210-L417-S0485	S0148-L308-S0036-L156-S2210-L417-S0485
出行 6	S0087-L021-S0088-L231-S0427-L097-S3676	S0087-L454-S3469-L209-S3676

◆ 以费用最少为目标：

● 目标函数：

$$\text{Min} \quad M_{sd}=\{M_{sd}^{N}\}, \quad N=0,1,2,3$$

● 函数说明：

当N=0，若$l_{ss} \cap l_{sd} \neq \varnothing$，则$t_{sd}^{0}$=1或依照所选线路中$S_S$与$S_d$间距离而定；否则$t_{sd}^{0}$ =∞。

当N=k(k=1,2,3)，M_{sd}^{k}=min{ M_{sd}^{0}+ M_{nd}^{k-1}} ，k为第一次换乘的车站号，可以从S_S向后沿各个线路顺次任意取站点。可用迭代的方法求出所有的M_{sd}^{k}。

● 算法简述：

求解最小费用的算法与求解最小时间的算法类似，只是在计算费用的时候要将公交线路分为单一票价和分段计价两类：单一票价全程都为1元，分段计价每20站加收1元。

类似问题一中的模型二，给出换车上限N=3。在N=0，1，2，3分别采用遍历搜索的方法，对N取各个值时统计每种乘车方法花费的车票费，从中找出花费最少的方法。

故算法具体步骤省略不述。

● 模型求解及分析：

用matlab7.0编程进行搜索(程序见附录)，结果如下：

表 3：以时间为目标函数的求解结果

	最少花费(单位：元)					
换乘次数	出行 1	出行 2	出行 3	出行 4	出行 5	出行 6
0	∞	∞	∞	∞	∞	∞
1	3	∞	3	2	∞	2
2	3	3	3	3	3	3
任意换乘次数	3	3	3	2	3	2

说明：∞表示在该换乘次数下不存在一条线路可以到达终点站。

由最少费用的表格可以看出，在换乘次数为 0、1、2 时，最少花费不超过 3 元。若换乘次数 N≥3，至少上 4 次车，花费最少为 4 元，肯定不是最小花费。所以在求最少花费的路线时，换乘次数 N≥3 的情况不再考虑。

表 4：以费用最少为目标的乘车路线

	费用最少的乘车路线
出行 1	S3359-L436-S1784-L167-S1828
出行 2	S1557-L084-S0055-L348-S2361-L361-S0481
出行 3	S0971-L013-S2184-L417-S0485
出行 4	S0008-L159-S0400-L474-S0073
出行 5	S0148-L024-S0345-L140-S3037-L104-S0485
出行 6	S0087-L021-S0630-L231-S0427-L097-S3676

说明：满足费用最少的行车路线有多种，仅列其中的一种换乘次数最少的行车路线。

由于人们出行时，多数情况以尽快到达为首要目标。综合考虑时间和费用因素，列出用时最少路线的花费，最少换乘路线与最少花费的对照表格。

表 5：各种乘车需求下花费的费用对比

	出行 1	出行 2	出行 3	出行 4	出行 5	出行 6
所有路线中最少花费	3	3	3	2	3	2
最少换乘的乘车花费	3	3	3	3	3	3
用时最少的乘车花费	3	4	3	4	4	3

按照题中要求满足查询者的不同需要：由表 2 和表 4 可得到以时间最省，换乘次数最少，费用最省为目标的乘车路线。

由表 5 可知不同目标下的乘车花费相差不大，故以乘车时间和换乘次数为主要目标综合给出推荐最优路线。由表 1 可得，出行 1、3、6 的最短时间路线需换乘 2 次，且换乘 2 次比换乘 1 次大大节省了出行时间。故选换乘 2 次的路线。出行 2、4、5 的最短时间路线换乘 3 次。但换乘 3 次比换乘 2 次减少的时间不足 10 分钟，且换乘 2 次比换乘 1 次的出行时间有很大改善。考虑人们出行时间相差不大情况下希望换乘次数尽量少，故这三个问题也选择换乘 2 次的最短时间成车路线。

表6：各种出行推荐线路

	推荐的最佳乘车路线	出行时间	出行花费	换乘次数
出行 1	S3359-L015-S2903-L485-S1784-L167-S1828	64	3	2
出行 2	S1557-L084-S1919-L189-S3186-L460-S0481	106	3	2
出行 3	S0971-L013-S2517-L290-S2159-L469-S0485	103	3	2
出行 4	S0008-L043-S1383-L290-S2184-L345-S0073	67	3	2
出行 5	S0148-L308-S0036-L156-S2210-L417-S0485	106	3	2
出行 6	S0087-L021-S0088-L231-S0427-L097-S3676	46	3	2

6. 问题二

6.1 问题分析：

问题二把地铁加入了考虑范围。把地铁线路视为一种特殊的公交线路，将地铁站和公交站合并到一起作为站点总集$\{V\}=\{S\}\cup\{D\}$，地铁线路和公交线路合并到一起作为边集{E}，Δt_{ii+1}表示从站点V_i到相邻站点V_{i+1}所花的时间。若$V_i, V_{i+1}\in\{S\}$，Δt_{ii+1} =3；若$V_i, V_{i+1}\in\{D\}$时，Δt_{ii+1} =2.5。记公汽换乘地铁的时间为$t1$=6min,地铁换乘公汽的时间为$t2$=7min，在地铁站进行公汽换车的时间为$t3$=11min，t_{sd}为从起始站Ss到终点站S_d所花费的时间，t'_{sd}，t''_{sd}，t'''_{sd}为第一类、第二类、第三类线路起始站Ss到终点站S_d所花

费的时间。

考虑到时间仍为首要选择标准，根据子问题定理，把复杂的路线分解为求多个子问题的最优解，利用换车次数受限的通用模型求解。再将子问题最优解加和即为问题二全局最优解，

6.2 模型补充：

补充假设：

⑴两地铁站之间乘坐地铁的耗时最短。

⑵T1上下行正好相反，T2是双向的。

⑶下面讨论的换乘次数不包括T1和T2之间换乘。

为了便于分析，在搜索线路时，将所走的线路进行分类：第一类不经过地铁线；第二类经过地铁线只进行公汽换乘，不乘坐地铁；第三类乘坐地铁。第一类乘车线路和问题一的乘车路线是相同的，最短乘车时间即为问题一中的结果。对于第二类和第三类路线，下面通过几个定理对不同类的最短路所具有的特性进行描述：

✧ 定理二：如果S_s到S_d的最短路径R_{sd}属于第二类，那么在R_{sd}中有且仅有一次公汽换乘是在地铁站发生。

证明：如果R_{sd}有两次公汽换乘分别在D_p，D_q发生，那么根据补充假设⑴，可用D_p到D_q的地铁线路替换R_{sd}中D_p到D_q的路径，从而得到花费时间更少的路径，与R_{sd}是最短路矛盾。证毕。

✧ 定理三：如果S_s到S_d的最短路径R_{sd}属于第三类，那么在R_{sd}中仅进出地铁站一次。

证明：与定理三类似，依据补充假设⑴，如果进出地铁有两次以上的就会产生矛盾，细节从略。证毕。

6.3 模型建立：

对第一类、第二类、第三类线路建立三个子模型。

6.3.1 第一类的子模型：

这类线路与问题一中讨论的线路完全相同。这里不再讨论

6.3.2 第二类线路的子模型：

- 目标函数：

$$\mathrm{Min}\quad t''_{sd} = t(Si,S(D_i)) + t(S(D_i),Si) + t3$$

- 函数说明：

对第二类路线，设在一地铁站D_i进行公汽换站，这个地铁站可换乘的公汽站集合记为$S(D_i)$。根据定理三仅在地铁站D_i进行唯一一次换乘。于是有$t''_{sd} = t''_{sDi} + t''_{Did}$。记$t''_{sd}(D_i)$为第二类线路在地铁站$D_i$进行公汽换站时的总花费时间，$t(Si,S(D_i))$为站点$S_i$到

站点集 $S(D_i)$ 中某站点所花费的时间, $t(S(D_i),Si)$ 为站点集 $S(D_i)$ 中某站点到站点 S_i 花费的时间。若使乘车总时间 t''_{sd} 最小，它的子问题都应取到最小，那么问题就转化为求 $t(S_s,S(D_i))\big|_{\min}$ 和 $t(S(D_i),S_d)\big|_{\min}$。

- 算法步骤：

Step1:设 S_k 为集合 $S(D_i)$ 中任意一站点。换乘次数上限N=3不变，在地铁站经历了一次公汽换乘，Min t''_{sk} 可用问题一中基于换车次数受限的通用模型求解。

Step2:令 S_k 搜遍集合 $S(D_i)$，求出 $t(S_s,S(D_i))\big|_{\min}$。

Step3:同理求出 $t(S(D_i),S_d)\big|_{\min}$。

Step4:对于任意一个选定的进行公汽换乘的地铁站 D_i，记 $t''_{sd}(D_i)$ 第二类线路在地铁站 D_i 进行公汽换站时的总乘车时间。将各子问题结果相加即可求出 $t''_{sd}(D_i)$。

Step5:遍历搜索所有地铁站 D_i，求得 $t''_{sd}\big|_{\min}$ =Min{$t_{sd}''(D_i)$}

6.3.3　针对第三类路线的子模型：

- 目标函数：

$$\text{Min}\quad t'''_{sd} = t(S_s,S(D_i)) + \bar{t}_{ij} + t(S(D_j),S_d) + t1 + t2$$

- 函数说明：

对于第三类路线，设进入和离开地铁线的地铁站分别为 D_i， D_j，可换乘的公汽站点集为 $S(D_i)$，$S(D_j)$。$t(S_i,S(D_i))$ 为站点 S_i 到站点集 $S(D_i)$ 中某站点所花费的时间，$\bar{t}_{ij}$ 为地铁站 D_i 、 D_j 间的行驶时间。那么乘车总时间为 t_{sd} = $t(S_s,S(D_i))$ + $\bar{t}_{ij}$ + $t(S(D_j),S_d) + t1 + t2$ = $t(S_s,S(D_i)) + \bar{t}_{ij}$ + $t(S(D_j),S_d)$ +13。根据子问题定理，要求乘车总时间 t'''_{sd} 的最小值，即求 $\overline{t_{ij}}\big|min$、$t(Ss,S(D_i))\big|min$ 和 $t(S(Dj),Sd)\big|min$。

- 算法步骤：

Step1:固定 D_i D_j 后，由于仅有两地铁站可进行地铁换线，所以将 D_i D_j 间换乘上限设为两次, 用基于换车次数受限的通用模型求解，求出 $\overline{t_{ij}}\big|min$。

Step2: $t(Ss,S(D_i))\big|min$ 和 $t(S(Dj),Sd)\big|min$ 在第二类线路的问题中已经解出，可直接求得。

Step3: 对于选定的 (D_i, D_j)，记 $t_{sd}'''(D_i, D_j)$ 为在地铁站 D_i, D_j 进出地铁的线路乘车总时间，将各子问题最优解相加即可求出 $t_{sd}'''(D_i, D_j)\big|_{\min}$。

Step4: 遍历搜索全部的进出地铁站的站点对 (D_i, D_j)，比较得出第三类线路中的 $t_{sd}'''\big|min = \mathrm{Min}\{t_{sd}'''(D_i, D_j)\}$。

6.4　模型求解及结果分析：

最终目标函数:

$$t_{sd}\big|_{min} = \mathrm{MIN}\{t_{sd}'\big|_{\min},\ t_{sd}''\big|_{\min},\ t_{sd}'''\big|_{min}\}$$

用matlab7.0编程进行搜索求解(程序见附录)，求得的 $t_{sd}'\big|_{\min}$，$t_{sd}''\big|_{\min}$，$t_{sd}'''\big|_{min}$ 结果，经比较求出 $t_{sd}\big|_{min}$：

表7：各类乘车线路的最少乘车时间

	最少乘车时间					
线路选择	出行 1	出行 2	出行 3	出行 4	出行 5	出行 6
第一类线路	64	99	103	63	102	46
第二类线路	82	117	128	75	119	50
第三类线路	77	114	96	52.5	87.5	36.5
所有线路	64	99	96	52.5	87.5	36.5

分析上表得出，对出行1和出行2， 最优乘车路线属于第一类线路。对出行3，乘地铁仅会节省不足10分钟，又考虑到人们希望尽可能不换车，且乘地铁会另外花费3元，所以最佳路线不经过地铁，仍选第一类线路。那么出行1、2、3选择的线路就和问题一中不考虑地铁的情况完全相同。对于出行4、5、6，选第三类路线乘地铁会有较大的时间节省，因此为最佳路线。

从上表中还可以看出第二类路线，即在地铁站进行公汽换站的路线，所花费的时间最长，且这次换站会增加换乘站数和费用，故应尽量避免在地铁站进行公汽换站。对出行4、5、6，虽然乘地铁会节省很多时间，但会多花费3元，所以可根据实际要求选择省时乘地铁还是节省费用乘公交车。

仍以时间最省为首要目标，给出问题二考虑地铁后的最佳乘车路线。（对需乘地铁的线路仅给出起止车站和进出地铁的车站）

表8：各种出行推荐线路

	推荐的最佳乘车路线	乘车时间	乘车花费
出行 1	S3359-L015-S2903-L485-S1784-L167-S1828	64	3
出行 2	S1557-L084-S1919-L189-S3186-L460-S0481	99	3
出行 3	S0971-L013-S2517-L290-S2159-L469-S0485	96	3
出行 4	S0008-D30-D25-S0073	52.5	6
出行 5	S0148-D02-D21-S0485	87.5	6
出行 6	S0087-D29-D36-S3676	36.5	5

7. 问题三

7.1 问题分析：

假设我们得到了任意两站间的步行时间矩阵H。把公汽站和地铁站统一视为站点 V_i，构成点集 $\{V\}$。

引入了步行换站的方式后，问题变得更加复杂，求解也变得更加困难。在这里我们只将以做出的模型一和模型三应用于本问题，给出模型的描述而不进行求解。

7.2 模型一的应用:

将原有的拓扑图 $G(V,E)$ 变换，站点集合 $\{V\}$ 不变，扩充集合 $\{E\}$ 为 $\{\widetilde{E}\}$，$\{\widetilde{E}\}$ 使 $G(V,E)$ 变为完全图 $\widetilde{G}(V,\widetilde{E})$。$\forall$ $E_{ij} \in \{\widetilde{E}\}$，如果 $Eij \in \{E\}$，则 E_{ij} 的权值不变；否则 $E_{ij} = H_{ij}$。

在权值明确之后，类似问题中的模型一，对 $\widetilde{G}(V,\widetilde{E})$ 进行疏松处理得到 $\widehat{G}(V,\widetilde{E})$，任意两点之间的最短路就可以用 Djikstra 算法解决，疏松处理的方法与模型一中完全相同。特别的，如果 $V_i \in \{D\}$，即 V_i 是地铁站点，则不需要对其进行疏松处理。

$\widetilde{G}(V,\widetilde{E})$ 经过疏松处理之后，点集和边集都极大得扩充了，可以预见该模型的求解非常困难。

7.3 模型三的应用：

针对上述情况，建立分类子问题模型。把问题分为两类：第一类不经过步行换站。第二类经过步行换站。

- 模型分析：

  第一类：与问题二考虑的线路完全相同，可用相同方法得出。这里不再讨论。

  第二类：因为步行速度很慢，换站要花费很长的时间，为了满足出行时间少的目标。

- 补充假设：

  1）在整条线路中只进行一次步行换站。

  2）存在步行换站的时间上限 t_h。

那么在站点集 $\{V\}$ 中遍历搜索站点对 (V_i,V_j)，记他们的步行换站时间为 $t(V_i,V_j)$。把所有 $t(V_i,V_j) \le t_h$ 的站点对组成站点对集合 $\{(V_i,V_j)\}$。根据假设 1)，仅进行一次步行换站，换站的站点对就设为 (V_i,V_j)。

- 模型建立：

  目标函数为：

$$\text{Min} \quad t_{sd} = t_{SVi} + t(V_i,V_j) + t_{Vjd}$$

在选定(V_i,V_j)后，设$t_{sd}(V_i,V_j)|_{\min}$为经选定的站点对(V_i,V_j)步行换站花费的最小时间，$t(V_i,V_j)$为一已知的值。由子问题定理得：

$$t_{sd}(V_i,V_j)|_{\min}=t_{SVi}|_{\min} + t(V_i,V_j)+t_{Vjd}|_{\min}$$

在剩下的站中不进行步行换站。求解$t_{SVi}|_{\min}$，$t_{Vjd}|_{\min}$的问题类型与问题二的相同，可用问题二的方法直接求出。

再令站点对遍历跑遍站点对集$\{(V_i,V_j)\}$，通过比较求得

$$t_{sd}|_{\min}=Min\{t_{sd}(V_i,V_j)|_{\min}\}$$

8. 模型灵敏度分析

在此，我们只关注主要优化目标--时间的变化：

8.1 公交车行驶时间变化分析：

我们把所有公交线路所需的行驶时间在原来的基础上增加和减少&t 后，用原模型求得此条件下的 6 个出行问题中前 3 问的最优时间$\widehat{t_{sd}}$，观察行驶时间对最优时间的边际影响$\Delta t_i=|\widehat{t_{sd}}-t_{sd}|$和换乘次数的影响。结合多方因素考虑&t 从±0.1—0.9 以步长为 0.2 取值，结果如下：

表 9：Δt_i-&t

&t	0.1	0.3	0.5	0.7	0.9	-0.1	-0.3	-0.5	-0.7	-0.9
△t1	1.8	5.4	9	11.2	13.3	1.8	5.4	9	12.6	16.2
△t2	3.9	11.7	19.5	27.3	35.1	3.9	11.7	21.7	31.5	41.3
△t3	3.1	9.3	15.5	21.7	25.9	3.1	9.3	15.5	21.7	27.9

在&取正值和负值时绘制两张图表：

图 3：随&t 增加的Δt_i-&t 图像

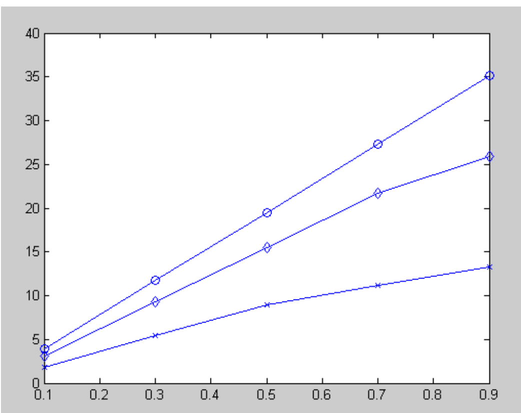
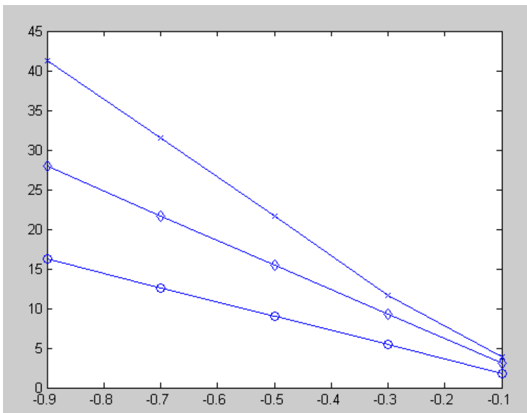


图 4：随&t 减少的Δt_i-&t 图像



8.2 换乘时间变化分析

同样的，所有公交与地铁所需的行驶时间在原来的基础上增加和减少$t，求得前三问最优解$\widetilde{t_{sd}}$，观察换乘时间对最优时间的边际影响$\partial t_i = |\widetilde{t_{sd}} - t_{sd}|$和换乘次数的影响，$t 从±0.1—0.1.7 以步长为 0.4,取值结果如下：

表 10：$t-$\partial t_i$

$t	0.1	0.5	0.9	1.3	1.7	-0.1	-0.5	-0.9	-1.3	-1.7
a t1	0.2	1	1.8	2.6	3.4	-0.2	-1	-1.8	-2.6	-3.4
a t2	0.3	1.5	2.7	3.9	5.1	-0.3	-1.5	-2.7	-3.9	-5.1
a t3	0.2	1	1.8	2.6	3.4	-0.2	-1	-1.8	-2.6	-3.4

该表绘制的图为一直线，即不会改变乘车路线。

1. 开始的时候$\Delta t_i = |\widehat{t_{sd}} - t_{sd}|$与&t 呈正比例关系，比例系数即最短线路经历的路线数。故可得结论：线路越长，公交车行驶时间变化对出行时间的影响越大。

2. $\partial t_i = |\widetilde{t_{sd}} - t_{sd}|$与$t 也呈正比例关系，比例系数即最短线路经历的换乘次数。故可得结论：线路换成次数越多，换乘时间变化对出行时间的影响越大。

3. 总体来看，公交车行驶时间变化的影响大于换乘时间变化的影响。

4. 由图 3，图 4，Δt_i、∂t_i的曲线发生了偏折，说明这时最短路径的选择发生了变化。观察图表可得结论：当公交车行驶时间变大时，最短路径可能改变为换乘较多的路线；当公交车行驶时间变小时，最短路径可能改变为换乘较少的路线。

8.3 对乘客的一些建议：

当发生如塞车等导致行驶时间变长的情况时，乘客应考虑更多的换乘路线，灵活乘车。而当车流量较少，车速较快时，换车会花费相对很多的时间，乘客应尽量避免换车。

9. 模型评价与改进方向

9.1 模型评价：

9.1.1 模型优点：

1） 通过假设舍弃了对问题影响不大的因素，只保留了乘车时间，换车次数，乘车花费这几个核心因素，使问题更加清晰，易于讨论。
2） 我们评价最优乘车路线时综合考虑了三种因素的影响，运用了多目标优化的思想。得出的线路更全面合理。
3） 相对传统的 Djikstra 算法,文中建立的基于换乘次数受限的通用模型大大提高了运算效率，缩短了运算时间，符合实际情况。
4） 在解决问题二时，我们把复杂问题分类讨论，并分成多个易于解决的子问题求解，使问题解决起来更简单明了。

9.1.2 模型缺点：

虽然相对传统精典的的 Djikstra 算法，我们自己建立的模型计算效率有相当提高，但计算量仍十分庞大，花费时间较长。而且找到的解会也不能保证是全局最优解。

9.2 模型改进方向：

9.2.1 遗传算法模型

针对已有模型在求解最优解时候效率低下的缺点进行改进，我们可以引入遗传算法模型。因为遗传算法具有收敛速度快，全局搜索能力好，所以比较适合用来解决非常困难的寻优问题。

- 编码方法：每个染色体代表一条出行路线，染色体长度可变。染色体 Ga 可表示为 $(S_1,S_2,S_3,S_4...S_K)$，若所求问题中的起始点是 S_S、目的地是 S_d，$S_1=S_S$，$S_k=S_d$。
- 适应函数：是因函数 F(Ga) 没有解释表达式，而是由 Matlab7.0 程序编写，返回值是 Ga 代表的路线所需的行车时间。
- 实施保优策略：将每一代按适应函数值升序排列，删除值最大的个体，用上一代至最小的个体代替，这样可以加快收敛速度。
- 复制策略：将每一代按适应函数值升序排列，值较小的获得较大的复制概率。
- 剪切、拼接及变异策略：使用随机的单点剪切，自由随机拼接和单点随机变异。

9.2.2 增加换乘上限：

针对模型二有可能得不到全局最优解的缺点，可以增加换乘次数 N 的上限。但要以求解效率的下降为代价，当上限过高时，效率可能还不及模型一用 D_{ij} 算法求解。

参考文献

[1]姜启源，数学模型（第三版）$[M]$，北京：高等教育出版社，2003。
[2]薛定宇等，高等应用数学问题的 matlab 求解，北京：清华大学出版社，2004。
[3]雷英杰等，matlab 遗传算法工具箱及应用，西安：西安电子科技大学出版社，2004。